Simple Vertex fitting algorithms

Rajendran Raja
15-Oct-2004

We present simple algorithms to determine the vertex of the interaction in MIPP. The fitted helices are used for these algorithms.

Simple algorithm (good for interaction trigger studies)

In this algorithm, the intersection of two tracks is found in the y-z view (they are straight lines). The circles (helix projections in the x-z view) are swum to this common intersection point. If the x co-ordinates of the two circles at the z intersection point are the same within some tolerance, then the helices are said to intersect and the x,y,z point of the intersection is taken as the vertex. The z co-ordinate of this pair is histogramed for each such track pair and should give us an idea of where the interactions are coming from. Finding the vertex this way avoids the need to find the intersection between two circles. One needs to allow for the distance of the found vertex from the TPC while making the tolerance cut, allowing a bigger tolerance the further away the vertex.

Vertex finding algorithm(for finding multiple vertices)

We find vertices in the y-z view by least squares fitting of straight lines. The vertex is assumed to be a two-dimensional point and one finds the distance of closest approach from this point to each straight line. One minimizes the sum of squares of the distance of closest approach from this vertex point to all the straight lines with respect to the vertex position. This yields the co-ordinates of the vertex. Then one cuts away tracks whose distance of closest approach exceeds a certain tolerance to the found vertex. One repeats the minimization and continues the iteration till the number of tracks associated with the vertex stabilizes. Then one swims the circles of the associated tracks to the vertex in this view and performs tolerance cuts as in the first algorithm.

One then proceeds to find secondary vertices using tracks that remain after the first vertex. Having found vertices in this fashion, three dimensional fit to the vertex can be done to the found tracks (perhaps using MINUIT and errors on all parameters.).

Mathematics formulae

We fit straight lines in the y-z view. The formulae for distance of closest approach and others are most elegantly expressed if we use complex numbers. So in what follows we will use the convention **z=x+iy**. So y will be vertical and x horizontal, pointing along the MIPP z axis.

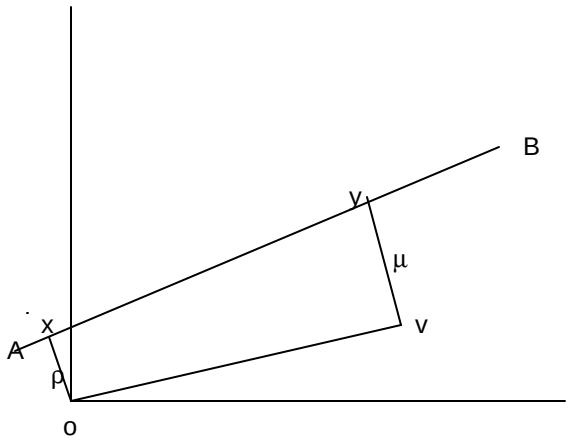

Figure 1 AB is a straight line in a co-ordinate system with O as the origin. V is a vertex point. The distance of closest approach of AB to V is given by the line VY of length m. The line OX of length r is the distance of closest approach of the line AB to O.

As shown in Figure 1, an arbitrary complex number z on the line AB is given by

$$z = ire^{iJ} + le^{iJ}$$

where the angle θ is the angle the straight line AB makes to the real axis and the first term is the complex number representing the line OX and λ is the distance from the point x to the complex number represented by z lying on AB. Let us denote the complex number OV by $ve^{i\phi}$. Then the complex number OY is given by

$$OY = ve^{if} + me^{iq} = ire^{iq} + le^{iq}$$

where the second equation comes from the condition that the point y lies on the straight line. One can solve for the distance of closest approach μ trivially.

$$m = r - il + ive^{i(j-J)}$$

Imposing the condition that μ is real yields

$$m = r - v\sin(j - q)$$
$$l = v\cos(j - q)$$

The intersection of two straight lines can be solved very quickly as

$$x = \frac{r_1 \cos q_2 - r_2 \cos q_1}{\sin(q_2 - q_1)}$$
$$y = \frac{r_1 \sin q_2 - r_2 \sin q_1}{\sin(q_2 - q_1)}$$

For the first algorithm the intersection (x,y) of two straight lines characterized by the pairs of numbers (ρ_1, θ_1) and (ρ_2, θ_2) is given by the above equations.

For the second algorithm, the distance of closest approach to μ to a straight line can be re-written in terms of the Cartesian co-ordinates v_x,v_y of the point v as

$$m = r - v_y \cos q + v_x \sin q$$

Then for a number of straight lines, the sum of squares of all the distances of closest approach can be written,

$$S^2 = \sum_{tracks} m_i^2 = \sum (r_i - v_y \cos q_i + v_x \sin q_i)^2$$

Minimizing S^2 with respect to $v_{x \text{ and } Vy}$ yields the set of equations,

$$Av_x + Bv_y = E$$

$$Cv_x + Dv_y = F$$

where,

$$A = D = -\sum \sin q_i \cos q_i$$

$$B = \sum \cos^2 q_i$$

$$C = \sum \sin^2 q_i$$

$$E = \sum r_i \cos q_i$$

$$F = -\sum r_i \sin q_i$$

These equations can be solved to yield

$$v_x = \frac{DE - BF}{AD - BC}$$

$$v_y = \frac{-CE + AF}{AD - BC}$$